

# 基于代码审查的综合软件质量评估方法

彭裕辉,王双喜

(中国地质大学 数理学院,湖北 武汉 430074)

**摘要:**评价一套软件系统的质量,一方面要对其所表现出来的外部特征进行研究,另一方面还应对其内部结构进行分析。因为构成软件系统的程序代码质量如何直接决定了整个软件系统的质量水平,所以程序代码中的缺陷分布特征也直接反映了软件系统的质量特征。介绍了如何通过抽样审查程序代码的方法建立软件缺陷分布函数,从而达到由软件缺陷分布来对软件质量进行综合、量化评估的目的。

**关键词:**软件质量评估;代码审查;缺陷分布

**中图分类号:** TP393.06

**文献标识码:** A

**文章编号:** 1672-7800(2010)03-0039-02

## 1 国内外研究现状

国内外针对软件质量评估提出了很多质量度量模型。1968年由 Rubey 和 Hurtwick 就软件的一些特性提出了度量方法,但尚未建立质量度量模型,所提出的度量方法也不完整。1976年,Boehm 等人提出了定量的评价软件质量的概念,给出了 60 个质量度量公式,首次提出了软件质量的 3 层模型。1978 年,Waters 和 McCall 提出了从软件质量要素、准则到度量的 3 层次式的软件质量度量模型。McCall 认为,软件的质量有 11 个要素构成,即正确性、可靠性、效率、完整性、可使用性、可维护性、可测试性、灵活性、可移植性、重复使用性和连接性。ISO 于 1985 年提出建议,软件质量度量模型由 3 层组成:高层软件质量需求评价准则、中层软件质量评价设计评价准则和底层软件质量度量评价准则。ISO3 层次模型来自 McCall 等人的模型,高层、中层、底层分别对应与 McCall 模型中的特性、质量准则和度量。

上海软件中心根据 ISO/ TC97/ SC7 的建议,同时参照 McCall 模型和 Boeing 模型,并结合我国实际情况综合构成了 SSC (Shanghai Software Center) 软件质量度量模型及度量方法,从而形成了 SSC 软件质量评价体系。

## 2 代码审查与评价

### 2.1 检查表设计

缺陷检查表是根据软件质量度量特征归类编写的一些通用的、良好编程规范或者编程注意事项,本文称之为评价原则,严格遵循这些原则进行编程能够保证代码质量的各种特性的

提高。缺陷检查表设计结构如图 1 所示。

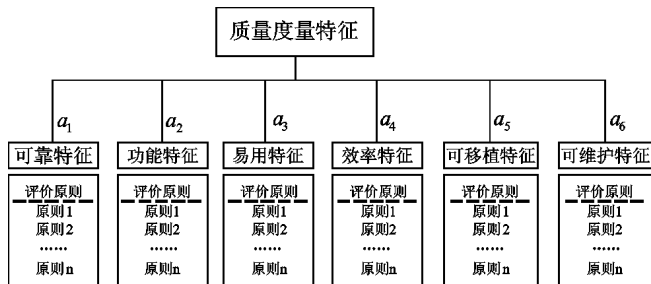


图 1 缺陷检查表设计结构

其中  $a_1, a_2, a_3, \dots, a_n$  为各项质量特性的权重。

每类质量特征包括多条评价原则,每条评价原则从程序代码的角度来阐述了程序应该遵循的编写规范,检查表设计时候应该注意尽量全面概括某一个质量特性的各项评价原则,保证同一质量特性中各条评价原则彼此独立、互不干扰或者重叠。

### 2.2 度量权重设计

对于不同的软件产品根据其性质的不同,质量衡量的标准也不同。例如国防武器系统和商用电子制表软件对其质量的要求是不同的,国防武器系统可能对安全性、易维护性要求更高;而商用电子指标软件可能对其可靠行、易用性、易移植性要求更高。因此,在评价一种软件产品之前首先要对每种度量特征规定相应的权值。

这里的权值只是一个相对的数字,例如:如果对可移植性要求只是一般,则可将可维护性权值设定为 3,如果对可靠性要求较高,可将其设定为 5,依次类推为每项质量特性根据具体情况设定相应的权值,最后得到权重矩阵  $A = [a_1, a_2, a_3, a_4, a_5, a_6]$ 。因此对于国防武器系统,电子商务系统可能的度量权重设计模式分别如表 1、表 2 所示。

**作者简介:**彭裕辉(1982-),男,湖北孝感人,中国地质大学(武汉)数理学院硕士研究生,研究方向为概率统计及应用;王双喜(1985-),男,湖北黄冈人,中国地质大学(武汉)数理学院硕士研究生,研究方向为概率统计及工程结构可靠性应用。

表1 国防武器系统度量权重设计表

| 质量度量特征 | 度量权重 | 相对重要程度 |
|--------|------|--------|
| 可靠特征   | 5    | 非常重要   |
| 功能特征   | 3    | 重要     |
| 易用特征   | 2    | 一般     |
| 可移植特征  | 3    | 重要     |
| 可维护特征  | 4    | 非常重要   |

表2 电子商务系统度量权重设计表

| 质量度量特征 | 度量权重 | 相对重要程度 |
|--------|------|--------|
| 可靠特征   | 5    | 非常重要   |
| 功能特征   | 3    | 重要     |
| 易用特征   | 4    | 非常重要   |
| 可移植特征  | 4    | 非常重要   |
| 可维护特征  | 2    | 一般     |

### 3 缺陷统计分析

#### 3.1 抽样审查

对待评估系统的全部代码进行抽样,例如从系统全部代码中随机抽取  $n$  份代码,每份代码包含  $m$  行代码(多余  $m$  行的可以从随机截取  $m$  行代码),这里  $n$  代码相当于  $n$  次抽样,显然,每组抽样缺陷数据都来自于同一样本空间。

对抽取的每份代码参照代码检查表进行逐行评审,找出每份代码中的缺陷,并将其按照软件度量特征进行归类记录,这样就可以得到  $n$  份抽样数据,每份抽样数据表明在固定的行代码中发现的各类缺陷数总数,对于第  $i$  份代码而言,假设对每类缺陷的累计计数分别为  $i = (i_1, i_2, i_3, \dots, i_6)$ ,其中  $i_k, 1 \leq k \leq 6$  是第  $i$  份代码第  $k$  类质量特性的最终缺陷总数,从而这份代码的综合缺陷贡献分值为  $r_i = \sum_{j=1}^6 w_j i_j$ ,全部这些通过评审得到的数据组成了一个样本集,记作  $R = (r_1, r_2, r_3, \dots, r_n)$ 。

通过将归类于某项度量特征的缺陷数与该度量特征类型的权重进行加权,  $w_j i_j$  从而达到区分软件关键质量特征的目的。

#### 3.2 求解缺陷分部函数

将每份代码的最终评价得分  $R$  作为随机变量,假设此随机变量  $R$  满足正态分布  $N(\mu, \sigma)$ ,利用中心距最大熵法可以求解该分布的参数  $\mu$ ,分布函数为  $F(x) = \int_0^x \frac{1}{\sqrt{2\pi}} e^{-\frac{(t-\mu)^2}{2\sigma^2}} dt$ ,  $0 < t < \infty$ 。在这里  $t$  为每份抽样代码中发现的缺陷数所贡献的分值,  $F(x)$  的实际意义代表全部抽样中每份样本代码中包含的缺陷所贡献的分值小于  $x$  的概率。

### 4 质量等级度量

引入评价集  $V_k = (v_{k1}, v_{k2}, v_{k3}, \dots, v_{ki})$ ,其中,  $v_{ki}$  由具体的软件质量测评部门根据软件的特性和实际经验给出,  $k$  为每行代

码规定的缺陷贡献分值的上限,  $v_{ki} (i = 1, 2, 3, \dots, n)$  对应着一个综合质量等级标准,  $v_{k1} < v_{k2} < v_{k3} < \dots, v_{ki}$ ,从  $v_{k1}, v_{k2}, v_{k3}, \dots, v_{ki}$  级别依次增加,等级越高证明软件综合质量越高。本文通过达标法对软件系统质量进行等级评判,具体评判方法如下。

若  $F(k) > v_{ki}$ ,则说明该抽样代码的质量标准达到了第  $i$  个级别,从而可以评定被检测系统代码质量等级最低为  $i$  级。

至此对软件系统的综合质量水平已给出了一个量化水平标准,通过建立的缺陷分部函数可以很容易地得出被测软件综合质量水平。

### 5 结束语

本文论述方法的优点在于从软件系统内部进行抽样分析判别,从而从实质上把握了软件的缺陷,具体有以下几个方面:能够除测试之外的另一个角度评估系统的质量,弥补了仅仅通过系统测试来评估软件系统质量的不足之处;可操作性强,评估结果在一定程度上量化了软件的综合质量,弥补了仅仅通过千行代码缺陷数的不足之处,为开发高质量的软件产品提供了必要的保证。

该方法的不足之处在于:基于代码缺陷分析的软件质量评估方法对工作的人员要求较高,要对程序设计和业务领域有比较深入的了解;本文对代码缺陷分布遵循的模型假定为正态分布,这满足一般的经验,一定程度上能够满足应用,但是还缺乏理论验证;通过本文提出的方法最终得到的是可靠性特征、效率特征、可维护性特征和可移植性特征等几个软件质量方面的综合评价,而没有对某一个方面给出具体的量化参数。

#### 参考文献:

- [1] 朱三元. 软件质量及其评价技术[M]. 北京:清华大学出版社,1990.
- [2] ISO/IEC 9126 - - 1: Information Technology - Software Product Quality - Part 1: Quality Model. ISO/IEC JTC1/SC7/WG6,Junio, 1998.
- [3] 陈军,马溪骏. 软件质量审查技术及其进展[J]. 电子质量,2004(1).
- [4] 孙梦,宋晓秋,巢翌. 软件程序代码质量度量技术研究[J]. 计算机工程与设计,2006(2).
- [5] 袁玉宇. 一个实用的软件质量评估模型[J]. 计算机工程,2003(5).
- [6] Deutsch M, Willis R. Software Quality Engineering. Englewood Cliffs, NJ: Prentice - Ha H, 1988.
- [7] Boehm B W, Brown J R. Quantitative evaluation of software quality [C]. 2nd International Conference on software Engineering, IEEE Computer Society, 1979.
- [8] 刘峻,原林. 软件质量度量研究及其分析[J]. 开发研究与设计技术, 2007(18).
- [9] 邢大红,曹佳冬,汪和才,等. 软件度量学综述[J]. 计算机工程与应用,2001(1).

(责任编辑:杜能刚)